Do Not Perform Arithmetic with Unvalidated Input

William L. Fithen, Software Engineering Institute [vita[3]]

2005-10-03

Careless modulo arithmetic can introduce vulnerability.

Description

According to [Seacord 05]:

> Integers represent a growing and underestimated source of vulnerabilities in C and C++ programs. This is primarily because boundary conditions for integers, unlike other boundary conditions in software engineering, have been intentionally ignored. Most programmers emerging from colleges and universities understand that integers have fixed limits, but because these limits were either deemed sufficient, or because testing the results of each arithmetic operation was considered prohibitively expensive, violating integer boundary conditions has gone almost entirely unchecked in commercial software.

For an indepth coverage of this issue in C and C++, see Safe Integer Operations[6].

References

[Blexim 02]	blexim. *Basic Integer Overflows*. http://www.phrack.org/phrack/60/p60-0x0a.txt (2002).
[Hoglund 04]	Hoglund, Greg & McGraw, Gary. *Exploiting Software: How to Break Code.* Boston, MA: Addison-Wesley, 2004.
[Horovitz 02]	Horovitz, Oded. *Big Loop Integer Protection*. http://www.phrack.org/phrack/60/p60-0x09.txt (2002).
[Howard 03a]	Howard, Michael. *Reviewing Code for Integer Manipulation Vulnerabilities*. http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dncode/html/secure041020 (2003).
[Seacord 05]	Seacord, Robert C. *Secure Coding in C and C++*. Boston, MA: Addison-Wesley, 2005.
[Thompson 05]	Thompson, Herbert & Chase, Scott. *The Software Vulnerability Guide*. Charles River Media, 211-222. 2005.

SEI Copyright

3. daisy:320 (Fithen, William L.)

6. daisy:312 (Safe Integer Operations)

Velden

Naam	Waarde
Copyright Holder	SEI

Velden

Naam	Waarde
is-content-area-overview	false
Content Areas	Knowledge/Guidelines
SDLC Relevance	Implementation
Workflow State	Publishable

1. http://www.sei.cmu.edu/about/legal-permissions.html